

Final Gathering on GPU

Toshiya Hachisuka
The University of Tokyo
toshiya@bee-www.com

1. Introduction

Producing global illumination image without noise is difficult because it requires a large number of samplings. Some approaches, such as photon mapping [Jensen 1996], can render noise-free image, but often cause low frequency biased noise. To eliminate this, the final gathering process regards the photon map as a rough global illumination solution, and then gathers illumination data to obtain definitive incident radiance on the visible intersection point [Jensen 1996]. Since the final gathering process is essentially the same as sampling a large number of rays on upper hemisphere of visible point, it is very costly to process at all visible pixels. The method described here is a fast final gathering method accelerated by using a GPU.

2. Proposed method

The method proposed here uses a randomly sampled global ray direction [Szirmay-Kalos 1998]. Using this direction, expensive ray intersection can be regarded as a multi-layered parallel projection (see Figure 1). Since graphics hardware can process matrix transformation including parallel projection very fast, such a method is natural and efficient compared with using GPU as a parallel ray tracing processor. Intersection points of global ray direction from sampling points cannot be obtained by single parallel projection, so inverse of depth peeling [Everitt 2001] is used instead to get actual intersection points. Each depth layer is projected from far to near and intersection points sample nearest depth layer. Depth layer texel behind the intersection point is culled by a KILL operation. To reduce the number of depth peeling iterations, criteria based on the number of sampled points can be used to stop the iteration without significant error on resulting image. Since the proposed method is per-pixel, it can process any form of problem that requires a large number of samples on hemisphere such as pre-computation of radiance transfer. To apply this method to a global illumination problem, some pre-computed rough global illumination solution of the scene is needed so I use grid photon map as a pre-process in the demo.

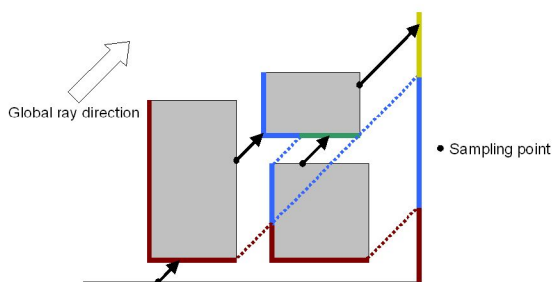


Figure 1: Relation between depth layer and final gathering. Shooting rays from each sampling points correspond to obtaining nearest points on visible depth layer.

3. Results

I implemented global illumination renderer using the proposed method. Figure 2 and 3 show the resulting images. The image resolution is 512 by 512. Note that it can be rendered within a few tens of seconds even though it samples a very large number of global directions on all visible pixels. All scenes were rendered using a Pentium4 2.8GHz and ATI Radeon 9700 Pro. The demo of global illumination renderer using this method is

available at <http://www.bee-www.com/parthenon/index.htm>. Since current implementation of a KILL operation does not stop the execution of fragment shader, future hardware with an early KILL operation will accelerate the proposed method further.

4. Conclusion

I presented implementation of final gathering on GPU. This method exploits coherency of gathering rays and fast parallel projection using GPU. Since this calculation uses matrix transformation of GPU, it is very fast compared with final gathering using ray shooting. In addition, it is a per-pixel sampling method so it can process any sets of points including ray-traced pixels, vertex data, light map textures.

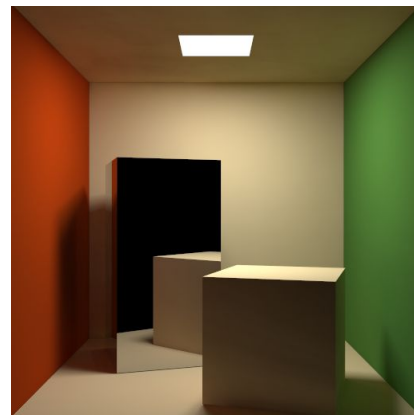


Figure 2: Cornell box with 1000 final gathering samples. Rendering time is 62 sec. (Final gathering takes 50 sec.)

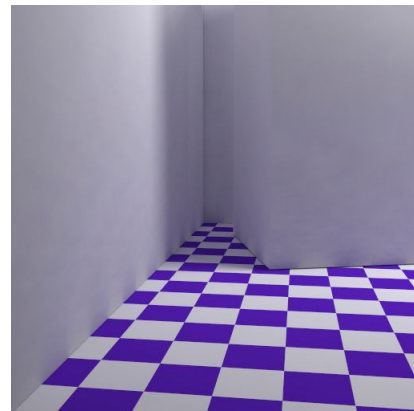


Figure 3: Close-up of textured Cornell box with 2000 final gathering samples (Final gathering takes 101 sec). Note that per-pixel inter-reflection from textured floor was rendered.

5. Reference

- Jensen, H. W. 1996. Global illumination using photon maps. *Rendering Techniques '96 (Proceedings of the Seventh Eurographics Workshop on Rendering)*, pages 21–30.
- Everitt, C. 2001. Interactive order-independent transparency. *Technical report*, NVIDIA Corporation.
- Szirmay-Kalos, L. 1998. Global Ray-bundle Tracing. *Technical Report/TR-186-2-98-18*.